Дрождин В.В., Жуков М.В. Древовидная развертка вычислительного автомата. // Проблемы информатики в образовании, управлении, экономике и технике: Сб. статей VIII Всерос. научно-техн. конф. – Пенза: ПДЗ, 2008. – С. 19-25.

ДРЕВОВИДНАЯ РАЗВЕРТКА ВЫЧИСЛИТЕЛЬНОГО АВТОМАТА

В.В. Дрождин, М.В. Жуков

Пензенский государственный педагогический университет
им. В.Г. Белинского,
г. Пенза

Постановка задачи. Известно, что отношение (функциональной) эквивалентности алгоритмов (программ) не является рекурсивно перечислимым, и, следовательно, проблема эквивалентных преобразований для них даже в самой слабой постановке [1] имеет отрицательное решение. В настоящей работе построен гомоморфизм, отображающий вычислительный автомат, предложенный в [2], в вычислительное дерево (ВД). Отображение автомата в ВД будем называть разверткой. Таким образом, для выявления изоморфизма автоматов достаточно показать изоморфизм их деревьев, тогда задача эквивалентных преобразований автоматов сводится к задаче эквивалентных преобразований соответствующих им деревьев [3].

Основные определения и принципы построения ВД. Рассмотрим автоматы с точки зрения функционального подхода. Каждый автомат задается функцией вида

$$\bar{y} = f(\bar{x}),$$

где $\bar{x} = \{x_1, .., x_k, x'_{k+1}, ...x'_m\}$ – множество входных переменных (сигналы от рецепторов);

$\bar{y} = \{y_1, .., y_l, y'_{l+1}, ...y'_n\}$ – множество выходных переменных (сигналы поступающие на эффекторы);

$x_i, y'_j \in C$ – информационные переменные;

$x'_i, y'_j \in (C \cup \theta)$ – γ-переменные;

C – множество сигналов;

θ – nul-сигнал;

m, n – размеры входного и выходного векторов (количество рецепторов, и эффекторов автомата) соответственно;

k, ℓ – количество информационных переменных.

Входные γ-переменные соответствуют сверхчувствительным рецепторам, выходные γ- переменные – слабоактивным эффекторам.

Далее будем рассматривать только простые автоматы, так как любой элементарный автомат является простым, а каждый сложный автомат раскладывается на совокупность простых автоматов, соответствующих обособленным элементам.

Пусть B = {f_1, ..., f_k} – базис f, где f_1, ..., f_k – автоматы вложенные в f. На B зададим произвольное отношение частичного порядка ρ (например отношение ≤ над идентификаторами автоматов в реестре). Введем вспомогательные функции: g(f), g'(f), e(f), e'(f) – возвращающие число входных, γ-входных, выходных и γ-выходных переменных функции f. Состоянием $\bar{z}$ автомата f с базисом B, на котором задано отношение порядка ρ, являются элементы множества

$$Z = S^{r(f) + r(f1) + \ldots + r(fk) + e(f)},$$

где S = {0, θ, 1}, 0 – соответствует невозбужденному состоянию рецептора или эффектора, θ, 1 – на них подан θ, не θ сигнал. Таким образом, состояние f описывается состоянием рецепторов f, состоянием рецепторов вложенных автоматов, расположенных в соответствии с ρ, и состоянием эффекторов f. Иногда (для краткости) будем выбрасывать все 0 – элементы, а для восстановления рецептора или эффектора, для которого указано состояние, состояние будем записывать в виде $\bar{z} = (x_l^i \ldots y_m)$, где i = 0...k и определяет номер автомата из B, где 0 – сам автомат f; ℓ – номер рецептора автомата i; m – номер эффектора автомата f. Элементы x или y из $\bar{z}$ находятся в стоянии 1, а элементы x' и y' – в θ либо в 1. Если вместо x или y указан θ, то соответствующий элемент находится в θ состоянии.

Состояние автомата может изменяться в следующих случаях:

1) при подаче сигналов на рецепторы автомата f;

2) после передачи сигналов от эффекторов отработавшего автомата на эффекторы f или на рецепторы связанных с ним автоматов.

Напомним, что если θ-сигнал передается со сверхчувствительного рецептора или эффектора на обычный рецептор или эффектор, то они не возбуждаются. Любые рецепторы или эффекторы, которые участвовали в возбуждении автомата, переходят в 0-состояние. При развертке автомата будем использовать также промежуточное состояние z_e^{fi} – состояние эффекторов автомата f_i.

Каждой неконечной вершине ВД автомата f может быть приписан некоторый предикат p ∈ P. Используются предикаты двух типов: унарный предикат состояния Ps, проверяющий, находится ли f в заданном состоянии; предикат выбора направления Pd, аргументом которого является стратегия выбора направления (если при построении автомата используется одна стратегия, то предикат – 0-арный). Из любой вершины выходят либо две дуги, одна из которых называется 0-дугой, а другая 1-дугой, либо ни одной, тогда вершина называется конечной, которая может быть активной, либо пассивной (тупик). Конечная вершина является активной, если все эффекторы f возбуждены, в противном случае вершина пассивна. При имитации вычислительного процесса вычисление из вершины v идет по 1-дуге, если приписанный ей предикат истинен, в противном случае – по 0-дуге.

Для построения ВД необходим алгоритм определения всех возможных активных стояний. Активное состояние – состояние, при котором, в зависимости от контекста, автомат переходит в возбужденное состояние (для этого все рецепторы должны быть возбуждены) либо рефлексирует (возбуждены все эффекторы). Пусть задан автомат f с $\bar{x} = \{x_1, .., x_k, x'_{k+1}, ...x'_m\}$. Для перехода f в возбужденное состояние существует 2^{m-k} активных состояний. Действительно, переменные 1...k остаются неизменными, а каждая из m-k γ-переменных может принимать один из сигналов: θ-сигнал или не θ-сигнал (х-сигнал). Поэтому получаем 2^{m-k} возможных состояния, которые будем обозначать через A_Z. Если рецепторы f соединены непосредственно с телом автомата, каждое состояние из A_Z приводит к общему результату – возбуждению f. В этом случае рассматривать отдельно эти состояния не имеет смысла, и множество активных состояний редуцируется до множества с единственным состоянием $\bar{z} = (x_1^0 ... x_k^0 x'^0_{k+1} ... x'^0_m)$ (П1). В случае, когда f сложный автомат и сигналы передаются от рецепторов f на рецепторы вложенных автоматов либо от внутренних эффекторов, то множество активных состояний сокращается по правилу (П2):

для каждого сверхчувствительного рецептора i проверяется, все ли связанные с ним рецепторы являются сверхчувствительными. Если да, то среди всех состояний выбираются пары, у которых не равны значения состояний только i-го рецептора автомата f. Каждая из таких пар заменяется соответствующим обобщающим элементом, в котором значение i-го рецептора равно x'^0_i.

На множестве активных состояний определим отношение $\rho_{\le}$. Предварительно дадим следующие определения:

состояние i является частью состояния j, если любой рецептор или эффектор, возбужденный в состоянии i, возбужден и в состоянии j;

длиной состояния j (|j|) будем называть суммарное количество возбужденных в нем рецепторов и эффекторов.

Тогда, $\rho_{\le} = \{(i, j) \in A_z^2 |$ i – есть часть j; если i не часть j, то $|i| \le |j|\}$. $\rho_{\le}$ – отношение частичного порядка, что позволяет задать упорядоченное множество состояний $A_Z^{\le} = (A_Z, \rho_{\le})$.

Алгоритм развертки. Построение ВД осуществляется в режиме имитации вычислительно процесса. При этом на каждом шаге определяются все возможные варианты дальнейшего хода вычислений. Условия выбора того или иного пути вычислений становятся вершинами дерева, а вычислительные операции приписываются дугам. Пусть задан автомат f с базисом B, графом связей автоматов G и начальным состоянием $\bar{z}_0 = (0)$.

Постоим множество активных состояний A_Z для f, минимизированное с помощью правил П1 и П2. В сложном автомате f рецепторы могут быть соединены либо с рецепторами автоматов из B, либо с эффекторами f, поэтому для каждого состояния $a \in A_Z$ рецепторов f в соответствии с G вычисляется состояние рецепторов автоматов из B и эффекторов f , после чего рецепторы f переводятся в невозбужденное состояние. Полученное множество состояний упорядочим $\rho_{\le}$ и обозначим как $A_{Z0}^{\le}$. Корнем ВД станет предикат $Ps(a_{max})$, где a_{max} – максимальный элемент множества $A_{Z0}^{\le}$.

$A_{Z0}^{\leq} = A_{Z0}^{\leq} / a_{max}$.

Пусть $Ps(a_{max})$ – ложь, то от $Ps(a_{max})$ проводится 0-дуга, и из $A_{Z0}^{\leq}$ снова выбирается максимальный элемент, и процедура повторяется. Когда $A_{Z0}^{\leq}$ станет пустым, то 0-дуга будет вести в пассивную конечную вершину.

Если значение $Ps(a_{max})$ – истина, тогда от $Ps(a_{max})$ проводим 1-дугу, и a_{max} станет $\bar{z}_1$ состоянием f. Если все эффекторы f находятся в возбужденном состоянии, то к концу 1-дуги приписываем активную конечную вершину, иначе – предикат Pd.

Необходимость выбора направления возникает каждый раз, когда состояние $\bar{z}$ может возбудить сразу несколько автоматов из B. Пусть B_z – упорядоченное множество автоматов, которые возбуждаются состоянием $\bar{z}$. Для определенности Pd, зададим как «выбран самый левый элемент множества B_z».

Если Pd – ложь, то из B_z убираем первый элемент, и процедура повторяется до тех пор, пока B_z не станет пустым (если $|B_z| \leq 1$, то предикат Pd можно не использовать). В этом случае и 0 и 1 дуги Pd соединяются с пассивными конечными вершинами.

Если Pd – истина, то 1-дуге приписывается выбранный автомат f_i, который переходит в возбужденное состояние, и все рецепторы, участвующие в его возбуждении, переходят в 0 состояние.

Далее весь процесс повторяется, только теперь активные состояния строятся для эффекторов автомата f_i (промежуточные состояния), в котором начальным состоянием выступает текущее состояние автомата f.

Заметим:

1) если 1-дуга от Pd ведет в конечную пассивную вершину, то Pd вместе с его дугами и завершающей вершиной можно удалить, а на его место поставить предикат, в который вела его 0-дуга.

2) если автомат содержит циклы, то ВД будет бесконечным.

При анализе ВД удобно пользоваться их графическим представлением. Для этого будем использовать следующие обозначения. Вершины будем изображать кружками, внутри которых вписан предикат. Заштрихованный кружок является активной конечной вершиной, а перечеркнутый – пассивной. Если вершина не имеет предшественников, то она является корнем ВД и помечается входящей стрелкой. 1-дуги будем помечать знаком '+'. Дугам, несущим функциональную нагрузку, приписываются символы функций из B.

Пример построения ВД. Процесс построения ВД рассмотрим на примере автомата f, реализующего функцию n! и заданного схемой на рис. 1 (для краткости всем элементам, кроме тел автоматов, присвоены уникальные числовые идентификаторы):

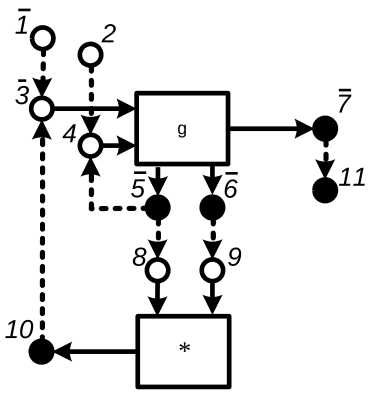

Рис. 1

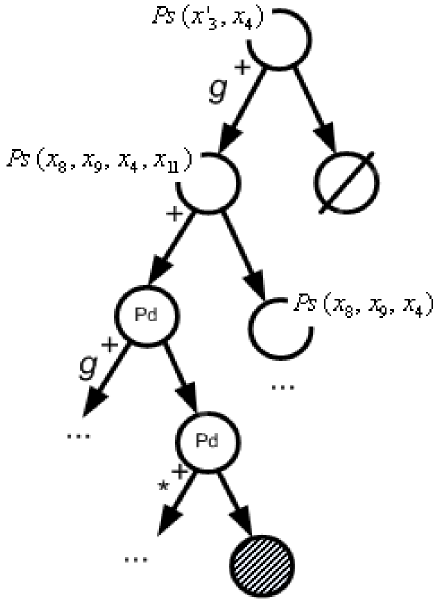

Рис. 2

1). Множество активных состояний для f состоит из $A_Z = \{(x_1, x_2), (\theta_1, x_2)\}$. Проводя сокращение по П2, получим $A_Z = \{(x'_1, x_2)\}$.

2). $A^{\leq}_{Z0} = \{(x'_3, x_4)\}$, добавляем предикат $Ps(x'_3, x_4)$ и перейдем по 1-дуге – $\overline{z_1} = (x'_3, x_4)$.

3). B_Z – состоит из одного элемента, поэтому предикат Ps опускаем, а 1-дуге $Ps(x'_3, x_4)$ приписываем g.

4). Строим множество промежуточных активных состояний эффекторов g (представим его в матричном виде, где строка задает состояние) – $Z^g_e = \begin{array}{r|l} 5 & \theta\theta\theta\theta1111 \\ 6 & \theta\theta11\theta\theta11 \\ 7 & \theta1\theta1\theta1\theta1 \end{array}$. По Z^g_e, $\bar{z}_1$, и, учитывая, что рецепторы 3 и 4 перешли в невозбужденные состояния, получаем $A^{\leq}_{Z1} = \begin{array}{r|l} 8 & 00010111 \\ 9 & 00101011 \\ 4 & 00010111 \\ 11 & 01001101 \end{array}$. Предикаты Ps с состояниями (0,0,0), (0,1,0), (1,0,0) будут вести в пассивное конечное состояние, поэтому их рассматривать не будем. Максимальным элементом множества $A^{\leq}_{Z1}$ является (1,1,1), поэтому следующей вершиной станет предикат $Ps(x_8, x_9, x_{11})$.

5). 0-дуга вершины $Ps(x_8, x_9, x_4, x_{11})$ переводит в предикат $Ps(x_8, x_9, x_4)$; 1-дуга – в предикат выбора направления.

6). Проводя аналогичные рассуждения, получим бесконечную развертку автомата.

Фрагмент построенного дерева приведен на рис. 2.

Библиографический список

1. Янов, Ю.И. О проблеме эквивалентных преобразований // Mitt. Math. Gesellsch. Der DDR 2-3 (1973), 47-58.

2. Дрождин, В.В. Построение систем моделирования на основе самоорганизующейся информационной среды // Материалы Междунар. науч.-техн. конф. «Модели и алгоритмы для имитации физико-химических процессов». – Таганрог: Изд-во НП «ЦРЛ», 2008. – С. 251 – 255.

3. Янов, Ю.И. Предельно полная система локальных правил эквивалентных преобразований программ // Discrete Math. Banach center publication, volume 7, Warsaw 1982, 75 – 103.